Probabilistic Reasoning with an Enzyme-Driven DNA Device

Iñaki Sainz de Murieta and Alfonso Rodríguez-Patón

Departamento de Inteligencia Artificial,
Universidad Politécnica de Madrid (UPM),
Campus de Montegancedo s/n, Boadilla del Monte 28660 Madrid, Spain
inaki.sainzdemurieta@upm.es, arpaton@fi.upm.es

Abstract. We present a biomolecular probabilistic model driven by the action of a DNA toolbox made of a set of DNA templates and enzymes that is able to perform Bayesian inference. The model will take single-stranded DNA as input data, representing the presence or absence of a specific molecular signal (the evidence). The program logic uses different DNA templates and their relative concentration ratios to encode the prior probability of a disease and the conditional probability of a signal given the disease. When the input and program molecules interact, an enzyme-driven cascade of reactions (DNA polymerase extension, nicking and degradation) is triggered, producing a different pair of single-stranded DNA species. Once the system reaches equilibrium, the ratio between the output species will represent the application of Bayes' law: the conditional probability of the disease given the signal. In other words, a qualitative diagnosis plus a quantitative degree of belief in that diagnosis. Thanks to the inherent amplification capability of this DNA toolbox, the resulting system will be able to to scale up (with longer cascades and thus more input signals) a Bayesian biosensor that we designed previously.

1 Introduction

Dynamic DNA nanotechnology is one of the areas of biomolecular computing that has developed most over the past decade. Many different models of DNA processors have been implemented since Adleman's seminal work [1]. We can find examples of DNA automata driven by restriction enzymes [2], deoxyribozyme-based DNA automata [3,4], DNA polymerase-based computers [5] or strand displacement circuits [6,7,8,9,10,11,12].

Most of the above models are designed as "use once" devices. This is a consequence of their operating principle: a set of molecules in a non-equilibrium state undertaking reactions and conformational changes until they reach a practically irreversible equilibrium state. Although this feature seems to be consistent with the objectives of structural DNA nanotechnology (e.g. DNA origami [13]), when we move to dynamic DNA nanotechnology the "use once" feature is a drawback rather than an advantage. Although they can still have very interesting applications (e.g. *in vitro* sensors and genetic diagnosis), every computation would

require a new DNA device. In order to achieve more complex behaviors, such as bistability or oscillations, biomolecular computing models need to be driven by a continuous input flux of energy [14]. This could be achieved, for example, by the DNAzyme-driven [3,4] and catalytic enzyme-free [10,11] models cited above, as long as there is a continuous supply of input ribonucleated strands and fuel strands, respectively, in the environment (e.g. in an open reactor). The design of other biomolecular computing models depends fundamentally on the existence of an input energy flux. For example, RNA computers work with a continuous supply of NTP, used by RNA polymerase as fuel in the transcription process [15,16].

DNA polymerase was one of the first computational primitives used in the early models of DNA computing [1,17]. It was therefore not surprising to find it in the first autonomous DNA computer model: the Whiplash machine [5]. However, after that milestone, DNA polymerase-driven models remained outside the mainstream for years, mainly due to the need for thermal cycles. Interest in this topic rekindled after some breakthroughs exploiting isothermal DNA amplification protocols [18], such as an improved Whiplash model [19] or the DNA toolbox developed by Rondelez's team [20,21,22].

The DNA toolbox is specially interesting due to its similarities with RNA computers: it is also driven by a continuous supply of NTP, which is used to extend input DNA strands and produce output strands. It has recently led to impressive achievements, such as reliable oscillations [20], bistability [21] or population dynamics models like predator-prey [22]. Its operation is based on the action of a set of enzymes (DNA polymerase, an isothermal DNA nicking enzyme and a single-strand specific exonuclease) on the input strands and a set of single-stranded DNA templates, enabling the following set of basic reactions (see Figure 1):

- *Polymerization and nicking.* After the hybridization of an input DNA strand $\overrightarrow{A}$[1] at the 3' end of a DNA template $\overleftarrow{AB}$, DNA polymerase produces the double strand $\overleftrightarrow{AB}$. Since the duplex $\overleftrightarrow{A}$ contains the recognition sequence of the nicking enzyme, the newly polymerized strand is cleaved in two fragments $\overrightarrow{A}$ and $\overrightarrow{B}$, which will dissociate from the template due to their shorter length. $\overrightarrow{B}$ can also be displaced by further DNA polymerase activity. As result of this process, the input strands $\overrightarrow{A}$ periodically generate new strands $\overrightarrow{B}$ (see left panel in Figure 1).
- *Inactivation.* A special type of input DNA strand $\overrightarrow{B}$ can be used to inactivate a template $\overleftarrow{DE}$. $\overrightarrow{B}$ does not fully bind the recognition sequence of the nicking enzyme in the template, and since it is longer than the regular inputs $\overrightarrow{D}$, $\overrightarrow{B}$ wins the competition to bind the template almost irreversibly. Moreover, its 3' end does not bind the template, avoiding the action of DNA polymerase (see right bottom panel in Figure 1).

[1] A DNA strand denoted $\overrightarrow{A}$ is supposed to be Watson-Crick complementary to a DNA strand denoted $\overleftarrow{A}$, and would form a duplex $\overleftrightarrow{A}$ when both molecules hybridize.

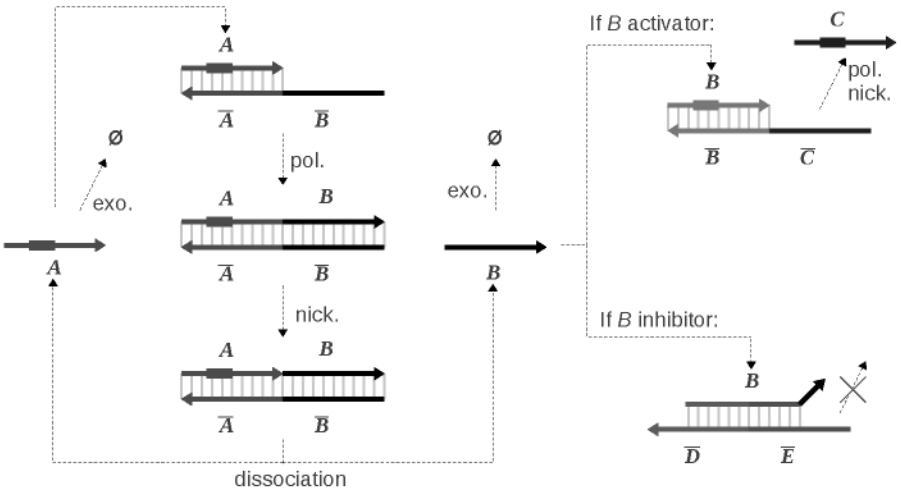

Fig. 1. DNA toolbox. The left panel shows the basic catalytic operation of the toolbox: the input strand $\overrightarrow{A}$ binds the 3' end of the DNA template $\overleftarrow{AB}$, allowing DNA polymerase to extend it forming the duplex $\overleftrightarrow{AB}$. Then the enzyme nickase binds to its recognition sequence in $\overleftrightarrow{A}$ (bold line) and cleaves the newly polymerized upper strand $\overrightarrow{AB}$ in two fragments $\overrightarrow{A}$ and $\overrightarrow{B}$, which can either dissociate from the template due to their shorter length, or let $\overrightarrow{B}$ be displaced by a new DNA polymerization of $\overrightarrow{A}$. The right panel shows the two possible operating modes of the output $\overrightarrow{B}$: as an activator it will enable the polymerization of another DNA strand $\overrightarrow{C}$ (see the motif at the top); as an inhibitor, it would bind in the middle of a DNA template $\overleftarrow{DE}$, inhibiting nicking and polymerization. All the DNA strands except the templates are subject to periodic degradation (see arrows pointing to ϕ).

– *Degradation.* Species dynamically generated by DNA polymerase are degraded by a single-strand specific exonuclease. DNA templates are protected from the action of the exonuclease thanks to DNA backbone modifications at their 5' end.

Inspired by recent works presented above by Rondelez's team, we have identified their DNA toolbox as an alternative to implementing probabilistic reasoning, which can be used when we want to consider diagnostic accuracy or uncertainty of tests in our clinical decisions (i.e., classic systems like Mycin [23]). With the aim of designing a model that can process this uncertainty, this article presents a Bayesian biosensor that reasons probabilistically and whose output represents the probability (value between 0 and 1) of a disease. Such a device can be used to estimate and update the probability of any diagnosis based in the light of new evidence, i.e., the presence or absence of a new specific signal (or set of signals). The DNA sensor device encodes two different probabilities as program data: the conditional probability of the signal given the disease ($P(signal|disease)$) and the prior probability of the disease ($P(disease)$). Then, when the sensor inter-

acts with an input representing the evidence of a signal (its presence or absence), Bayes' law is autonomously computed by means of enzymatic reaction cascades, releasing a set of DNA species whose concentration ratio encodes the posterior probability of the disease given the input ($P(disease|signal)$). We presented a similar model in [24], which used DNA strand displacement instead of Rondelez's DNA toolbox.

The rest of the chapter is structured as follows. Section 2 includes an example of Bayesian inference that can be performed with the model. Sections 3, 4 and 5 show the encoding of input signals and prior and conditional probabilities, respectively. Section 6 details how the model implements the Bayesian inference process. Finally, Section 8 summarizes the conclusions and future work.

2 Example of Bayesian Inference

This section describes a basic Bayesian inference example.

Let us imagine that we want to diagnose whether a patient is affected by a certain disease d, whose possible diagnosis is "disease present" (D_1) or "disease absent" (D_0).

Based on empirical data, we can know upfront the prior probability of the disease. For this example, we consider both diagnoses to be equiprobable, which is represented as follows:

$P(d) = \langle P(D = present), P(D = absent)\rangle = \langle P(D_1), P(D_0)\rangle = \langle 0.5, 0.5\rangle$.

Studying already diagnosed cases of this disease and its symptoms s (working as input signals), we can also ascertain upfront the conditional probability of a certain symptom (or signal) s given the disease d, $P(s|d)$:

$P(S = absent)|D = absent) = P(S_0|D_0) = 0.7$
$P(S = present)|D = absent) = P(S_1|D_0) = 0.3$
$P(S = absent)|D = present) = P(S_0|D_1) = 0.2$
$P(S = present)|D = present) = P(S_1|D_1) = 0.8$.

Now we test whether the patient has symptom s, which we interpret as a confirmation that the signal s is present (S_1). In the light of this new evidence, we can update our knowledge on the probability of the disease being present given that the signal is present, $P(D_1|S_1)$, applying the Bayes' law:

$$P(d|s) = \frac{P(s|d) \cdot P(d)}{P(s)} = \alpha \cdot P(s|d) \cdot P(d). \quad (1)$$

Since we do not know the prior probability of the signal $P(s)$, we can apply the second derivation of Bayes' law as stated in Equation 1:

$P(D_1|S_1) = \alpha \cdot P(S_1|D_1) \cdot P(D_1) = \alpha \cdot 0.8 \cdot 0.5 = \alpha \cdot 0.4$.

In order to find α, we need to calculate $P(D_0|S_1)$ as well:

$P(D_0|S_1) = \alpha \cdot P(S_1|D_0) \cdot P(D_0) = \alpha \cdot 0.3 \cdot 0.5 = \alpha \cdot 0.15$.

According to the foundations of probability theory, we know $P(D_1|S_1) + P(D_0|S_1) = 1$. We can use this knowledge to derive $\alpha = 1.81$ and $P(D_1|S_1) = 0.73$.

The biomolecular probabilistic inference devices described in the next sections of the paper can autonomously update their output probability values, such that they match the inference steps described in this example.

3 Encoding Input Evidences

Normally, a biomolecular device that senses real samples expecting a certain input signal In would reason as follows: if molecules In are present, the signal is present; otherwise the signal is absent. However, the devices that we propose use a different type of input logic, where the presence and absence of the signal are represented by the presence of different DNA species.

Thus, our input evidence is encoded using single-stranded DNA. A strand S_1 encodes the presence of an input signal, whereas a strand S_0 encodes the absence of the signal. As we are dealing with evidences, only one species can be present at a time: either S_1 (meaning the signal is present) or S_0 (meaning the signal is not present). These input signals will tell the sensor that the prior probability of the disease needs to be updated according to the given evidence.

However, if the system is to be able to deal with real biological samples, it needs to translate the presence of an external input signal In into strands S_1 (meaning input present in our system) and the absence of In into strands S_0 (meaning input absent in our system). A recent bistable implementation using this DNA toolbox illustrated an excellent way of translating the respective signals to produce strands S_1 and S_0 [21]. In this paper, a bistable switch producing a certain type of DNA species (which could be our S_0) in the absence of a certain type of input species In switched to producing another type of DNA species (which could be our S_1) in the presence of In. This model meets all the requirements to encode input evidence in the fashion described above. See [21] for details, which are omitted here due to space constraints.

4 Encoding Prior Probabilities

As illustrated by the example of Section 2, the prior probability of a disease is represented by the duple $P(d) = \langle P(D_1), P(D_0) \rangle$. Our model will use two different single-stranded DNA species to encode each possible probability value: $\overrightarrow{D_1}$ species representing $P(D = present)$ and $\overrightarrow{D_0}$ representing $P(D = absent)$. These strands will be produced from two DNA templates, $\overleftarrow{D_1 D_1}$ and $\overleftarrow{D_0 D_0}$. When $\overrightarrow{D_i}$ strands interact with their respective $\overleftarrow{D_i D_i}$ templates, $\overrightarrow{D_i}$ production increases (see Figure 2). At the same time, exonuclease degrades the production of $\overrightarrow{D_i}$ at a certain rate. The equations below govern this behavior:

$$\overrightarrow{D_i} + \overleftarrow{D_i D_i} \underset{k_a^{D_i} \cdot K_d^{D_i}}{\overset{k_a}{\rightleftharpoons}} \overrightarrow{D_i} \cdot \overleftarrow{D_i D_i} \xrightarrow{k_{cat}^{D_i}} \overrightarrow{D_i} + \overrightarrow{D_i} + \overleftarrow{D_i D_i} \tag{2}$$

$$\overrightarrow{D_i} \xrightarrow{k_{dec}^{D_i}} \phi\,, \tag{3}$$

where k_a is the association constant of $\overrightarrow{D_i}$, $K_d^{D_i}$ is the dissociation constant of $\overrightarrow{D_i}$ from the DNA template, $k_{dec}^{D_i}$ is the degradation rate of $\overrightarrow{D_i}$, and $k_{cat}^{D_i}$ is the rate of production of new strands $\overrightarrow{D_i}$. The constant really includes several reactions (polymerization, nicking and dissociation), but is confined to one here for reasons of space. Therefore, we would expect $k_{cat}^{D_i} << k_a^{D_i}$ and thus the respective Michaelis-Menten constant of the catalysis reaction would be $K_m^{D_i} \simeq K_d^{D_i}$ ($K_m^{D_i} = (k_a^{D_i} \cdot K_d^{D_i} + k_{cat}^{D_i})/k_a^{D_i}$).

When the system reaches equilibrium, the ratio between the concentration of both species will encode the prior probability, such that

$$P(D_i) = \frac{[\overrightarrow{D_i}]^{EQ}}{\sum_{i=0}^{1}[\overrightarrow{D_i}]^{EQ}} = \frac{[\overrightarrow{D_i}]^{EQ}}{\lambda}\,, \tag{4}$$

where λ represents the sum of $[\overrightarrow{D_0}]$ and $[\overrightarrow{D_1}]$ that encodes the maximum probability 1. Each equilibrium concentration $[\overrightarrow{D_i}]^{EQ}$ is a function of the initial concentration of the templates $\overrightarrow{D_iD_i}$. Section 6 shows the derivation of this function.

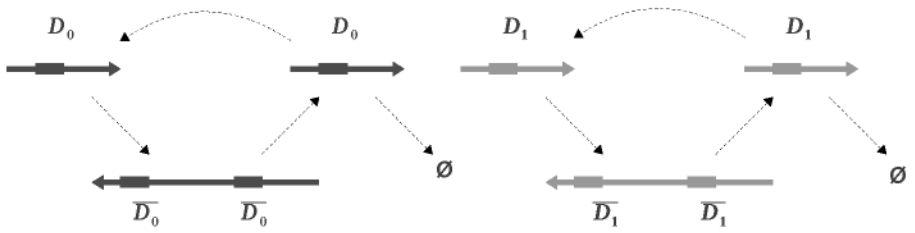

Fig. 2. Encoding prior probabilities. Thick regions of the strands represent the nickase recognition sequence. When a strand $\overrightarrow{D_i}$ at the top of the figure binds a template strand $\overleftarrow{D_iD_i}$ at the bottom of the figure, they form a complex $\overrightarrow{D_i}$:$\overleftarrow{D_iD_i}$ then DNA polymerase extends the upper strand to form the duplex $\overrightarrow{D_iD_i}$:$\overleftarrow{D_iD_i}$ and finally the enzyme nickase cleaves the newly polymerized strand in the middle. After the $\overrightarrow{D_i}$ strands dissociate from the template due to their short length, they can either be degraded by the exonuclease (arrows pointing to ϕ) or be recruited again by the template to produce more strands $\overrightarrow{D_i}$.

5 Encoding Conditional Probabilities

Conditional probabilities require the encoding of four different probability values: $P(S_0|D_0)$, $P(S_0|D_1)$, $P(S_1|D_0)$ and $P(S_1|D_1)$. Two different types of DNA templates will be used in the encoding of each probability value (see left side of Figure 3):

- Templates with format $\overleftarrow{D_i : D_i \wedge S_j}$ produce species $\overrightarrow{D_i \wedge S_j}$ in the presence of input strands $\overrightarrow{D_i}$ (see Figure 3), such that when the system reaches equilibrium $[\overrightarrow{D_i \wedge S_j}]^{EQ}$ is a function of $[\overrightarrow{D_i}]^{EQ}$ and $[\overleftarrow{D_i : D_i \wedge S_j}]$. The relative

concentration of the templates with format $\overleftarrow{D_i : D_i \wedge S_j}$ encodes each conditional probability value, such that:

$$P(S_j|D_i) = \frac{\beta_{ij} \cdot [\overleftarrow{D_i : D_i \wedge S_j}]}{\sum_{j=0}^{1} \beta_{ij} \cdot [\overleftarrow{D_i : D_i \wedge S_j}]} \tag{5}$$

$$\sum_{j=0}^{1} \beta_{ij} \cdot [\overleftarrow{D_i : D_i \wedge S_j}] = \sum_{j=0}^{1} \beta_{kj} \cdot [\overleftarrow{D_k : D_k \wedge S_j}] = \gamma \,,\ k \neq i \,, \tag{6}$$

where β_{ij} is a normalization coefficient and γ is the total normalized concentration of strands $[\overleftarrow{D_i : D_i \wedge S_j}]$ that represents probability 1. Section 6 will show the meaning of the β_{ij} coefficients and how $[\overrightarrow{D_i \wedge S_j}]^{EQ}$ is proportional to the product of $[\overrightarrow{D_i}]^{EQ}$ and $[\overleftarrow{D_i : D_i \wedge S_j}]$.

– Templates with format $\overleftarrow{D_i \wedge S_j : D'_i}$ have a twofold objective. First, they generate the output species D'_i, whose relative concentration will encode the posterior probability of the disease given the signal ($P(d|s)$). Second, in conjunction with the input signal species S_i, they select what posterior probability computation should be produced as output: when the input signal is S_1 (S_0), it binds and inactivates the strands $\overleftarrow{D_i \wedge S_0 : D'_i}$ ($\overleftarrow{D_i \wedge S_1 : D'_i}$) (see the crossed-out arrows in Figure 3), so that there is only one source of species D'_1 and another of D'_0, whose ratio will conform the output probability: the posterior probability $P(D_i|S_j)$ of the disease. All the templates with format $\overleftarrow{D_i \wedge S_j : D'_i}$ must have the same concentration, so that there are no changes of relative proportions from $[\overrightarrow{D'_i}']$ in relation to their respective source $[\overrightarrow{D_i \wedge S_j}]$.

The equations below govern the behaviour of these components:

$$\overrightarrow{D_i} + \overleftarrow{D_i : D_i \wedge S_j} \underset{K_d^{D_i} \cdot k_a^{D_i}}{\overset{k_a}{\rightleftharpoons}} \overrightarrow{D_i} \cdot \overleftarrow{D_i : D_i \wedge S_j} \xrightarrow{k_{cat}^{D_i \wedge S_j}} \overrightarrow{D_i} + \overleftarrow{D_i : D_i S_j} + \overrightarrow{D_i \wedge S_j} \tag{7}$$

$$\overrightarrow{D_i \wedge S_j} + \overleftarrow{D_i \wedge S_j : D'_i} \underset{K_d^{D_i \wedge S_j} \cdot k_a^{D_i \wedge S_j}}{\overset{k_a}{\rightleftharpoons}} \overrightarrow{D_i \wedge S_j} \cdot \overleftarrow{D_i \wedge S_j : D'_i} \xrightarrow{k_{cat}^{D'_i}} \ldots$$

$$\ldots \xrightarrow{k_{cat}^{D'_i}} \overrightarrow{D_i \wedge S_j} + \overleftarrow{D_i \wedge S_j : D'_i} + \overrightarrow{D'_i}' \tag{8}$$

$$\overrightarrow{S_k} + \overleftarrow{D_i \wedge S_j : D'_i} \underset{K_d^{S_k} \cdot k_a^{S_k}}{\overset{k_a}{\rightleftharpoons}} \overrightarrow{S_k} \cdot \overleftarrow{D_i \wedge S_j : D'_i} \,,\ k \neq j \tag{9}$$

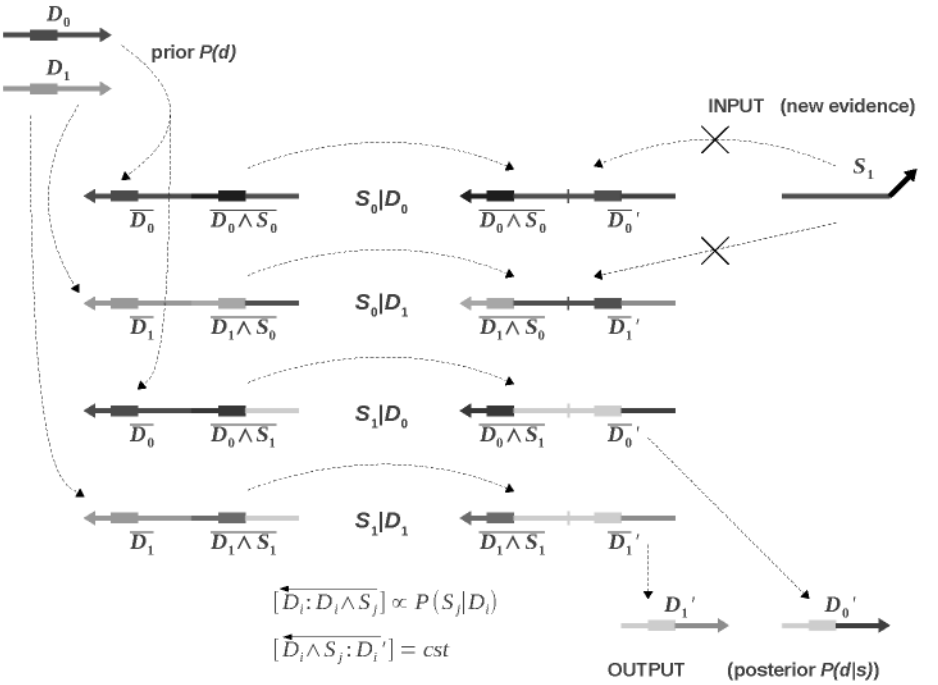

Fig. 3. Encoding conditional probabilities. The prior probability strands $\overrightarrow{D_i}$ bind the templates on the left side, enabling the production of $\overleftarrow{D_i \wedge S_j}$ strands (via polymerization and nicking) used to encode conditional probability. These strands will then activate the templates on the right side not protected by the input strands S_j (see the crossed-out arrows), producing the output strands $\overrightarrow{D_i'}$, whose concentration ratio encodes the posterior probability $P(d|s)$.

$$\overrightarrow{D_i \wedge S_j} \xrightarrow{k_{dec}^{D_i \wedge S_j}} \phi \tag{10}$$

$$\overrightarrow{D_i'} \xrightarrow{k_{dec}^{D_i'}} \phi \tag{11}$$

$$\overrightarrow{S_j} \xrightarrow{k_{dec}^{S_j}} \phi\,, \tag{12}$$

where k_a is the association rate of $\overrightarrow{D_i}$, $\overrightarrow{D_i \wedge S_j}$ and $\overrightarrow{S_k}$; $K_d^{D_i}$, $K_d^{D_i \wedge S_j}$ and $K_d^{S_k}$ are their respective dissociation constants; $k_{cat}^{D_i S_j}$ and $k_{cat}^{D_i'}$ are the production rates of strands $\overrightarrow{D_i \wedge S_j}$ and $\overrightarrow{D_i'}$; $k_{dec}^{D_i \wedge S_j}$, $k_{dec}^{D_i'}$ and $k_{dec}^{S_j}$ are the degradation constants of $\overrightarrow{D_i \wedge S_j}$ and $\overrightarrow{D_i'}$ and S_j.

6 Inference Process

6.1 Inference Steps

A high-level description of the inference process follows:

Goal. Update the concentration of $\overrightarrow{D_i'}$ strands once a new signal ($\overrightarrow{S_0}$ or $\overrightarrow{S_1}$) is detected.

Initial set-up. Add templates $\overleftarrow{D_i D_i}$ (whose concentration is a parameter in the encoding of prior probabilities), and templates $\overleftarrow{D_i : D_i \wedge S_j}$ and $\overleftarrow{D_i \wedge S_j : D_i'}$ (whose concentrations are parameters in the encoding of conditional probabilities).

Step 1. Add some $\overrightarrow{D_i}$, such that templates $\overleftarrow{D_i D_i}$ bring the production of strands $\overrightarrow{D_i}$ to its equilibrium concentration $[\overrightarrow{D_i}]^{EQ}$, which will be proportional to the prior probability $P(D_i)$.

Step 2. The $\overrightarrow{D_i}$ species bind the templates $\overleftarrow{D_i : D_i \wedge S_j}$, activating the production (via polymerization and nicking) of $\overrightarrow{D_i \wedge S_j}$ strands, whose equilibrium concentration $[\overrightarrow{D_i \wedge S_j}]^{EQ}$ is proportional to the conditional probability $P(S_j|D_i)$.

Step 3. The newly created "conditional probability strands" $\overrightarrow{D_i \wedge S_j}$ bind the templates $\overleftarrow{D_i \wedge S_j : D_i'}$ that are not protected by $\overrightarrow{S_0}$ or $\overrightarrow{S_1}$, activating the production (via polymerization and nicking) of the output species D_i'.

Read-out. The new concentration ratio of $\overrightarrow{D_i'}$ encodes the posterior probability $P(D_i|S_j)$.

This description is refined below providing a more thorough analysis of the process with estimations and derivations.

6.2 Modeling the Inference

From the equations presented in Sections 4 and 5, we can build a derivation that relates the output concentrations $[\overrightarrow{D_i'}]$ to the initial concentrations of the strands encoding prior and conditional probabilities.

Based on Equations 2, 3 and the Michaelis-Menten model [25], we can infer how $[\overrightarrow{D_i}]$ changes in time (see Equation 13) and, applying the equilibrium condition ($d[\overrightarrow{D_i}]/dt = 0$), obtain derivations for $[\overrightarrow{D_i}]^{EQ}$ (see Equation 14) and the initial $[\overleftarrow{D_i D_i}]$ (see Equation 15):

$$\frac{d[\overrightarrow{D_i}]}{dt} = \frac{k_{cat}^{D_i} \cdot [\overrightarrow{D_i}] \cdot [\overleftarrow{D_i D_i}]}{K_d^{D_i} + [\overrightarrow{D_i}]} - k_{dec}^{D_i} \cdot [\overrightarrow{D_i}] \tag{13}$$

$$[\overrightarrow{D_i}]^{EQ} = \frac{k_{cat}^{D_i}}{k_{dec}^{D_i}} [\overleftarrow{D_i D_i}] - K_d^{D_i} \tag{14}$$

$$[\overleftarrow{D_i D_i}] = \frac{k_{dec}^{D_i}}{k_{cat}^{D_i}}([\overrightarrow{D_i}]^{EQ} + K_d^{D_i}) \,. \tag{15}$$

A similar procedure can be applied for $\overrightarrow{D_i \wedge S_j'}$ from Equations 7 and 10, obtaining a derivation for $[\overrightarrow{D_i \wedge S_j'}]^{EQ}$ (see Equations 16 and 17). We are assuming $K_d^{D_i} >> [\overrightarrow{D_i}]$, which could be achieved with an appropriate temperature increase:

$$\frac{d[\overrightarrow{D_i \wedge S_j'}]}{dt} = \frac{k_{cat}^{D_i \wedge S_j}}{K_d^{D_i}}[\overrightarrow{D_i}] \cdot [\overleftarrow{D_i : D_i S_j}] - k_{dec}^{D_i \wedge S_j} \cdot [\overrightarrow{D_i \wedge S_j'}] \tag{16}$$

$$[\overrightarrow{D_i \wedge S_j'}]^{EQ} = \frac{k_{cat}^{D_i \wedge S_j}}{K_d^{D_i} \cdot k_{dec}^{D_i \wedge S_j}}[\overrightarrow{D_i}] \cdot [\overleftarrow{D_i : D_i S_j}] \,. \tag{17}$$

The formulation of $[\overrightarrow{D_i'}]$ is a bit more intricate, because the Michaelis-Menten derivation needs to consider the interaction of the inhibiting input species S_i, which represses the catalysis. Based on Equations 8, 9 and 11, and also assuming $K_d^{D_i \wedge S_j} >> [\overrightarrow{D_i \wedge S_j}]$, we can infer $[\overrightarrow{D_i'}]^{EQ}$ (see Equations 18 and 19):

$$\frac{d[\overrightarrow{D_i'}]}{dt} = \sum_{j=0}^{1} \frac{k_{cat}^{D_i'} \cdot [\overrightarrow{D_i \wedge S_j'}] \cdot [\overleftarrow{D_i \wedge S_j : D_i'}]}{K_d^{D_i \wedge S_j} \cdot (1 + \frac{[S_k^{k \neq j}]}{K_d^{S_k}})} - k_{dec}^{D_i'} \cdot [\overrightarrow{D_i'}] \tag{18}$$

$$[\overrightarrow{D_i'}]^{EQ} = \sum_{j=0}^{1} \frac{k_{cat}^{D_i'}}{K_d^{D_i \wedge S_j} \cdot k_{dec}^{D_i'} \cdot (1 + \frac{[S_k^{k \neq j}]}{K_d^{S_k}})}[\overrightarrow{D_i \wedge S_j'}] \cdot [\overleftarrow{D_i \wedge S_j : D_i'}] \,. \tag{19}$$

Taking into account that species S_0 and S_1 are never present at the same time, $[\overrightarrow{S_i}] >> [\overrightarrow{D_i \wedge S_j'}] + [\overleftarrow{D_i \wedge S_j : D_i'}]$ and $K_d^{S_k} << K_d^{D_i \wedge S_j}$, we can neglect the terms of the sum in Equation 19 where $[\overrightarrow{S_k'}] > 0$ and derive a simpler expression for $[\overrightarrow{D_i'}]$:

$$[\overrightarrow{D_i'}]^{EQ}_{[S_j]=0,\,[S_k^{k \neq j}]>0} = \frac{k_{cat}^{D_i'}}{K_d^{D_i \wedge S_j} \cdot k_{dec}^{D_i'}}[\overrightarrow{D_i \wedge S_j'}] \cdot [\overleftarrow{D_i \wedge S_j : D_i'}] \,. \tag{20}$$

Substituting Equation 17 in Equation 20, and reordering constant values to the left and variables to the right:

$$[\overrightarrow{D_i'}]^{EQ}_{[S_j]=0,\,[S_k^{k \neq j}]>0} = \frac{k_{cat}^{D_i'} \cdot k_{cat}^{D_i \wedge S_j} \cdot [\overleftarrow{D_i \wedge S_j : D_i'}]}{K_d^{D_i \wedge S_j} \cdot k_{dec}^{D_i'} \cdot K_d^{D_i} \cdot k_{dec}^{D_i \wedge S_j}}[\overleftarrow{D_i : D_i S_j}] \cdot [\overrightarrow{D_i}] =$$
$$= \beta_{ij} \cdot [\overleftarrow{D_i : D_i S_j}] \cdot [\overrightarrow{D_i}] \,. \tag{21}$$

In the above Equation 21, all the constant terms have been grouped in the parameter β_{ij} (already introduced in Equations 5 and 6). The term $[\overrightarrow{D_i}]$ is proportional to the prior probability (see Equation 4), and the product $\beta_{ij} \cdot [\overleftarrow{D_i : D_i S_j}]$ is proportional to the conditional probability (see Equation 5). The derivation below shows how the ratio between $[\overrightarrow{D_0}']^{EQ}$ and $[\overrightarrow{D_1}']^{EQ}$ determines posterior probability $P(d|s)$:

$$\frac{[\overrightarrow{D_i}']^{EQ}}{[\overrightarrow{D_0}']^{EQ} + [\overrightarrow{D_1}']^{EQ}} = \frac{\beta_{ij} \cdot [\overleftarrow{D_i : D_i S_j}] \cdot [\overrightarrow{D_i}]}{\beta_{0j} \cdot [\overleftarrow{D_0 : D_0 S_j}] \cdot [\overrightarrow{D_0}] + \beta_{1j} \cdot [\overleftarrow{D_1 : D_1 S_j}] \cdot [\overrightarrow{D_1}]} =$$

$$= \frac{\beta_{ij} \cdot \frac{\gamma}{\beta_{ij}} \cdot P(S_j|D_i) \cdot \lambda \cdot P(D_i)}{\beta_{0j} \cdot \frac{\gamma}{\beta_{0j}} \cdot P(S_j|D_0) \cdot \lambda \cdot P(D_0) + \beta_{1j} \cdot \frac{\gamma}{\beta_{1j}} \cdot P(S_j|D_1) \cdot \lambda \cdot P(D_1)} =$$

$$= \frac{P(S_j|D_i) \cdot P(D_i)}{P(S_j|D_0) \cdot P(D_0) + P(S_j|D_1) \cdot P(D_1)} = \frac{P(S_j|D_i) \cdot P(D_i)}{P(S_j)} = P(D_i|S_j) \,.$$

7 Discussion

The DNA biosensor presented here operates as a Bayesian inference device. It is capable of introducing quantitative information, highlighted by the molecular indicators or signals, into the tests. It builds on our previous work [24], but uses the DNA toolbox recently introduced by Rondelez [20,21] instead of the DNA strand displacement operation. Another aim was to map the basic concepts of probability theory and Bayesian inference into the toolbox motifs, for use as design patterns when implementing Bayesian reasoning with DNA.

The example detailed in Section 6 has used only one input signal. For this model to have realistic applications in genetic diagnosis, however, it needs to deal with more than one signal $(s^1, ..., s^n)$ for the same disease d (superscripts denote the signal number). According to Equation 1, the following formulation of Bayes' law would need to be solved: $P(d|s^1, ..., s^n) = \alpha \cdot P(d) \cdot P(s^1, ..., s^n|d)$. Assuming conditional independence of the signals given the disease (as in the naïve Bayes model [26]) we can derive the following expression: $P(d|s^1, ..., s^n) = \alpha \cdot P(d) \cdot P(s^1|d) \cdot ... \cdot P(s^n|d)$, meaning the initial probability statement with multiple input signals can be decomposed into conditional probability products, which can be encoded by cascading the devices presented here.

This research has addressed the two main improvement opportunities of the work that we presented elsewhere [24]:

Reusability. Devices are conceived for just one use. If the inputs are altered after the output signals become stable, the new output would not be correct any more. We would need a new initialised set of devices to deal with a new input. This research solves this problem with the action of the single-strand specific exonuclease, which periodically degrades all the non-template

strands not protected at their 5' end. This way, when the initial intput data flux S_k is stopped in favor of the new flux $S_{k'}$ ($k \neq k'; k, k' \in 0..1$), the system will converge to the total elimination of S_k (since that species is only degraded and not replenished). The same should happen with the intermediate species $\overleftarrow{D_i \wedge S_j}$ and output species $\overleftarrow{D'_i}$, driving the system to converge to a the correct output for input $S_{k'}$.

Signal attenuation. In theory, the model in [24] was also able to deal with multiple input signals by cascading the outputs as inputs of other conditional probability devices downstream. However, each inference iteration would attenuate the signal by an average of 50%. The replacement of strand displacement by an enzymatic catalysis, with inherent amplification capabilities, overcomes this drawback allowing longer inference cascades and thus more input signals.

8 Conclusions and Future Work

We have designed a biomolecular probabilistic expert system for genetic diagnosis. This is an enzyme-driven DNA device able to:

1. Encode diagnostic probabilistic information in single-stranded DNA.
2. Sense DNA inputs.
3. Process probabilistic information, encoded either as a steady state concentration of single-stranded DNA (for prior probabilities) or as a fixed concentration of single-stranded DNA (for conditional probabilities).
4. Release output molecules (duples of single-stranded DNA encoding a probability proportional to their concentration ratio).
5. Update the probability of the disease depending on the different single-stranded DNA inputs detected following Bayes' rule.

The model is autonomous and can be implemented according to the DNA toolbox presented in [20,21]. We think this and the other model that we introduced in [24] have the potential to deliver new quantitative applications of probabilistic genetic diagnosis *in vitro*. We plan to build, improve and generalize both models in a wet lab to work with all types of Bayesian networks (and not just naïve Bayes approaches [26]).

Acknowledgments. This research was partially supported by project BACTOCOM funded by a European Commission 7th Framework Programme grant (FET Proactive area) and by Spanish Ministry of Finance project TIN2012-36992.

References

1. Adleman, L.M.: Molecular computation of solutions to combinatorial problems. Science 266(5187), 1021–1024 (1994)
2. Benenson, Y., Gil, B., Ben-Dor, U., Adar, R., Shapiro, E.: An autonomous molecular computer for logical control of gene expression. Nature 429(6990), 423–429 (2004)
3. Stojanovic, M.N., Stefanovic, D.: A deoxyribozyme-based molecular automaton. Nature Biotechnology 21(9), 1069–1074 (2003)
4. Pei, R., Matamoros, E., Liu, M., Stefanovic, D., Stojanovic, M.N.: Training a molecular automaton to play a game. Nature Nanotechnology 5(11), 773–777 (2010)
5. Hagiya, M., Arita, M., Kiga, D., Sakamoto, K., Yokoyama, S.: Towards Parallel Evaluation and Learning of Boolean μ-Formulas with Molecules 48, 105–114 (1997)
6. Yurke, B., Turberfield, A.J., Mills, A.P., Simmel, F.C., Neumann, J.L.: A DNA-fuelled molecular machine made of DNA. Nature 406(6796), 605–608 (2000)
7. Seelig, G., Soloveichik, D., Zhang, D.Y., Winfree, E.: Enzyme-Free Nucleic Acid Logic Circuits. Science 314(5805), 1585–1588 (2006)
8. Rodríguez-Patón, A., de Murieta, I.S., Sosík, P.: Autonomous resolution based on DNA strand displacement. In: Cardelli, L., Shih, W. (eds.) DNA 17. LNCS, vol. 6937, pp. 190–203. Springer, Heidelberg (2011)
9. Sainz de Murieta, I., Rodríguez-Patón, A.: DNA biosensors that reason. Biosystems 109(2), 91–104 (2012)
10. Qian, L., Winfree, E.: Scaling up digital circuit computation with DNA strand displacement cascades. Science 332(6034), 1196–1201 (2011)
11. Qian, L., Winfree, E., Bruck, J.: Neural network computation with DNA strand displacement cascades. Nature 475(7356), 368–372 (2011)
12. Soloveichik, D., Seelig, G., Winfree, E.: DNA as a universal substrate for chemical kinetics. Proceedings of the National Academy of Sciences 107(12), 5393–5398 (2010)
13. Rothemund, P.W.K.: Folding DNA to create nanoscale shapes and patterns. Nature 440(7082), 297–302 (2006)
14. Kjelstrup, S., Bedeaux, D.: Non-Equilibrium Thermodynamics of Heterogeneous Systems. Series on Advances in Statistical Mechanics. World Scientific (2008)
15. Benenson, Y.: Synthetic biology with RNA: progress report. Current Opinion in Chemical Biology 16(3-4), 278–284 (2012)
16. Weitz, M., Simmel, F.C.: Synthetic in vitro transcription circuits. Transcription 3(2), 87–91 (2012)
17. Amos, M.: Theoretical and Experimental DNA Computation. Natural computing series. Springer, Heidelberg (2005)
18. Walker, G.T., Little, M.C., Nadeau, J.G., Shank, D.D.: Isothermal in vitro amplification of DNA by a restriction enzyme/DNA polymerase system. Proceedings of the National Academy of Sciences 89(1), 392–396 (1992)
19. Reif, J., Majumder, U.: Isothermal reactivating whiplash PCR for locally programmable molecular computation. Natural Computing 9, 183–206 (2010)
20. Montagne, K., Plasson, R., Sakai, Y., Fujii, T., Rondelez, Y.: Programming an in vitro DNA oscillator using a molecular networking strategy. Molecular Systems Biology 7(1) (2011)

21. Padirac, A., Fujii, T., Rondelez, Y.: Bottom-up construction of in vitro switchable memories. Proceedings of the National Academy of Sciences 109(47), E3212–E3220 (2012)
22. Fujii, T., Rondelez, Y.: Predator-prey molecular ecosystems. ACS Nano 7(1), 27–34 (2013)
23. Shortliffe, E.H., Buchanan, B.G.: A model of inexact reasoning in medicine. Mathematical Biosciences 23(3-4), 351–379 (1975)
24. Sainz de Murieta, I., Rodríguez-Patón, A.: Probabilistic reasoning with a bayesian DNA device based on strand displacement. In: Stefanovic, D., Turberfield, A. (eds.) DNA 2012. LNCS, vol. 7433, pp. 110–122. Springer, Heidelberg (2012)
25. Johnson, K.A., Goody, R.S.: The original michaelis constant: Translation of the, michaelis-menten paper. Biochemistry 50(39), 8264–8269 (1913)
26. Minsky, M.: Steps toward artificial intelligence. Proceedings of the IRE 49(1), 8–30 (1961)